Broadcasting and NP-completeness

Jean-Claude Bermond, Pierre Fraigniaud

BROADCASTING AND *NP*-COMPLETENESS

Jean-Claude BERMOND[1] and Pierre FRAIGNIAUD[2]

[1] 13 S, CNRS URA 1376
Université de Nice-Sophia Antipolis
bât 4, Rue A. Einstein
06560 Valbonne, FRANCE
(bermond@mimosa.unice.fr)

and

[2] LIP - IMAG, CNRS URA 1398
Ecole Normale Supérieure de Lyon
46 Allée d'Italie
69364 Lyon Cedex 07, FRANCE
(pfraign@lip.ens-lyon.fr)

Abstract

In this note, we answer two questions arising in broadcasting problems in networks. We first describe a new family of minimum broadcast graphs. Then we give a proof due to Alon of the *NP*-completeness of finding disjoint spanning trees of minimum depth, rooted at a given vertex.

1. Introduction

In the design and use of parallel computers, different elements are important. Among them are the topology of the interconnection network and the communication scheme. In this paper, we focus on one important communication problem:

> **Definition:**
>
> **Broadcasting** means sending a message from a given vertex to all other vertices in a network.
> The **initiator** is also called the root, and the broadcasting problem is also called OTA (One-To-All). ■

We consider the usual *store-and-forward* model for routing, in which a message that passes through intermediate nodes has to be stored in each intermediate processor before reaching its final destination. Two kinds of communication schemes are usually considered: **half duplex** and **full duplex**. In the half duplex mode, a link can be used at a given time in at most one direction; in the full duplex mode a link can be used simultaneously in both directions. Furthermore, we distinguish the **processor-bound model** (or 1-port model, or whispering) and the **link-bound model** (or shouting). In the first model, only one port can be used by a processor at a given time. In the second model, all the ports can be simultaneously used by a processor at any given time. There are many papers in the graph theory literature concerning this problem in the processor-bound model, and assuming that the communication cost is a constant (**constant model**). In parallel distributed memory architectures, it appears that the neighbor to neighbor communication time depends on a latency, or start up time β, and on a data transfer time per element, or propagation time, τ ($1/\tau$ is the bandwidth). Thus, sending a message of length L to a neighbor takes time $\beta + L\tau$ (**linear model**).

For more details on the results obtained in these different models, we refer to the two surveys of Hedetniemi, *et al.* **[1]** and Fraigniaud and Lazard **[2]**.

In this paper, we first give a short proof of a recent result in the constant model by giving an infinite family of minimum broadcast graphs. Then we consider the linear model, and show that it gives rise to new problems in graph theory; namely how to construct the maximum number of disjoint spanning trees of minimum depth rooted at a given vertex. In particular, we give a proof of the *NP*-completeness of this problem.

2. A Family of Minimum Broadcast Graphs

We consider the processor-bound and constant time model. Let G be a connected graph with n vertices, and let u be a message originator. We define the **broadcast time** of vertex u, b(u), to be the minimum number of time units required to complete broadcasting from vertex u. Note that $b(u) \geq \lceil \log_2 n \rceil$. We define the broadcast time of G, b(G), to be the maximum broadcast time of any vertex u in G.

We call a graph G a **minimum broadcast graph**, if it has the minimum number of edges among the graphs with n vertices and broadcast time $\lceil \log_2 n \rceil$. Let B(n) be the number of edges of a minimum broadcast graph.

It was conjectured in **[3]** that $B(2^k - 2) = (k-1)(2^{k-1} - 1)$. This has been recently proved by Khachatrian and Harutounian **[4]**, and Dinneen, *et al.* **[5]**. We present here a short proof, due to Monien, that points out that, in fact, Knodel **[6]** has constructed the desired graphs in its solution to a "gossiping" problem.

Theorem 1:

For any k, there exists a $(k-1)$-regular graph G with $2^k - 2$ vertices and broadcast time $b(G) = k$.

Proof:

Let G be a bipartite graph with two vertex sets, each of order $2^{k-1} - 1$. Vertex $(i,1)$ of the first part is connected to vertices $(i + 2^j - 1, 2)$, $j = 0,1,\ldots,k-2$ of the second part where all the integers are to be taken modulo $2^{k-1} - 1$. Clearly this graph has the required number of vertices and degree.

Call the edge from a vertex $(i,1)$ to the vertex $(i + 2^j - 1, 2)$, an edge of type j. A broadcast protocol is given as follows. At time j, $j = 1,2,\ldots,k-2$, each informed vertex sends the message along the edge of type $j-1$. At time k, each informed vertex, except the two that were first informed, sends the message along the edge of type 0. One can show by induction that at time j, 2^{j-1} consecutive vertices are informed in each part. So $b(G) = k$. ■

Corollary:

$B(2^k - 2) = (k-1)(2^{k-1} - 1)$.

Proof:

$B(2^k - 2) \leq (k-1)(2^{k-1} - 1)$ from the theorem above. Furthermore, if a graph G contains a vertex u of degree $k-2$ or less, at most $2^k - 3$ vertices can be informed in k units of time by a broadcast initiated at vertex u. Therefore, any minimum broadcast graph on $2^k - 2$ vertices has minimum degree at least $k-1$. ■

3. Disjoint Spanning Trees

We consider the link-bound and linear time model. We study both half and full duplex models. A half-duplex communication network is usually modeled by a graph G, and a full-duplex communication network by a symmetric digraph G*. We first give the theory for the full-duplex model.

Recall that, under the linear model, sending a message of length L to a neighbor takes time $\beta + L\tau$. The broadcast time of vertex u, b(u), is then the minimum time required for complete broadcasting from vertex u.

Let d(u,v) be the distance between the vertices u and v. We will denote by ecc(u) the **eccentricity** of vertex u, that is $\max_{v \in V} \{d(u,v)\}$. Let $m^+(S, V \backslash S)$ be the number of arcs going from S to V\S and let $c_G(u) = \min_{S \neq V, u \in S} \{m^+(S, V \backslash S)$. $c_G(u)$ can be regarded as the minimum number of arcs that must be deleted in order to make at least one vertex not reachable from u. Another interpretation of $c_G(u)$ is that there exist $c_G(u)$ arc disjoint paths from u to any vertex of G. Moreover, $c_G(u)$ is the maximum number of paths that are arc-disjoint (Menger's theorem).

We can obtain two different lower bounds by considering the total start-up time or the total data transfer time. First the broadcasting time b(u) is at least $ecc(u)\beta$. Consider now a subset S of V containing u such that $m^+(S, V \backslash S) = c_G(u)$, and let v be a vertex of V\S. The total bandwidth of the arcs between S and V\S is $c_G(u)/\tau$ and therefore the minimum time to send the message from u to v is at least $L\tau/c_G(u)$. In summary,

$$(1) \qquad b_G(u) - \max \{ecc(u)\beta, \frac{L}{c_G(u)}\tau\}$$

There exist different ways to perform broadcasting from an originator u. The efficiency of these protocols depends on the ratio $\beta/L\tau$ (see [7] for more details). In case of long messages, a classical technique is pipelining which allows a broadcast in

$$\{\sqrt{L\tau} + \sqrt{(ecc(u) - 1)\beta}\}^2$$

In case of a very long message, we can improve this time in finding p spanning trees rooted at u and that are pairwise arc disjoint. We cut the message into blocks, each of size L/p, and pipeline each block on a different spanning tree. Suppose the maximum depth of the spanning trees is h, then the broadcasting time is

$$(2) \qquad \left(\sqrt{\frac{L\tau}{p}} + \sqrt{(h-1)\,\beta}\right)^2 .$$

Thus, this theory gives rise to the following problem:

Problem 1:

Find, in any digraph, as many as possible arc-disjoint spanning trees rooted at a given vertex, and of maximum depth as small as possible. □

If we do not consider the depth, this problem is well known. For instance, we can use the following theorem of graph theory due to Edmonds **[8]** (see Lovász **[9]** for a short proof).

Theorem 2 (Edmonds' Theorem):

The maximum number of pairwise arc disjoint spanning trees rooted at a vertex u is equal to $c_G(u)$. ■

Applying this theorem on eqn(2), and comparing with eqn(1), shows that there exists an asymptotically optimal broadcasting protocol. Moreover, this protocol can be defined in a polynomial time since finding the $c_G(u)$ arc-disjoint spanning trees can be done in a polynomial time.

A similar theory can be developed for the half-duplex model and leads us to the following problem:

Problem 2:

In any graph, find as many as possible edge-disjoint spanning trees rooted at a given vertex, and of maximum depth as small as possible. □

These problems have been studied for particular interconnection networks like the hypercube **[10]**, the de Bruijn networks **[7]**, and the toroidal grids **[11]** **[12]**.

However, to the best of our knowledge, the complexity of problems (1) and (2) was unknown. This question was asked during Graph Theory Day 22, and was recently answered by Alon:

Theorem 3 (Alon):

The following problem is *NP*-complete:
INSTANCE: A graph G with a root u.
PROBLEM: Are there 2 edge-disjoint spanning trees in G of depth 2 rooted at u?

Proof:

The reduction is from the hypergraph 2-colorability problem which is: given a hypergraph $H = (V,E)$ decide if it is 2-colorable; that is, if there is a 2-coloring of the set of vertices V by red and blue so that in each edge there is at least one red and at least one blue vertex. This problem has been proved to be *NP*-complete by Lovász **[13]**.

Given such a problem, let us construct a graph G with a special vertex u and with depth 2 as follows. The set of vertices of G consists of three sets: one is only the root u; the second is the set V (corresponding to the vertices of the hypergraph H) as well as two additional vertices that we call r and b; and the third is the set E (corresponding to the edges of H). Now, u is connected to all the vertices in V and to r and b and to no other vertex. Vertices r and b are adjacent to all the vertices in V. Futhermore, r and b are connected by two parallel edges. In addition, every v in V is adjacent in G to all the vertices in E that represent the edges of H containing v in H.

<u>Remark</u>: If one wishes to avoid parallel edges between r and b, replace r by two vertices r_1 and r_2, and b by two vertices b_1 and b_2. Now make all four of these adjacent vertices adjacent to u and vertices of V, also add all the four edges (r_i, b_j) ;$i, j \in \{1, 2\}$.

It suffices to show that H is 2-colorable if and only if G has two edge-disjoint spanning trees of depth 2 rooted at u.

Suppose H is 2-colorable, take a proper 2-coloring and define the two trees as follows. The first tree (call it the red tree) consists of the edges from u to all the red vertices in V, together with one edge from each edge e in E to some red vertex in V which is adjacent to it (there is such a vertex as the coloring is a proper 2-coloring). Also, the edge (u,r) is in the red tree (r was for red) as well as the first edge (r,b) and edges from r to all the blue vertices of V. The second tree (the blue one) is defined similarly: edges from u to the blue vertices of V and to b, the second edge (r,b), edges from b to the red vertices in V, and also one edge from each e in E to some blue vertex in V. This gives two edge-disjoint trees of depth 2 rooted at u, as needed.

Suppose now that there are two edge disjoint trees of depth 2 rooted at u. Denote them by R and B. Let V_R be the set of all vertices v in V so that (u,v) is an edge of R, and let V_B be the set of all vertices v in V so that (u,v) is an edge of B. Since R and B are both spanning trees and both have depth 2, and since the only paths of length 2 in G between u and vertices of E are through a vertex of V, one easily observes that both V_R and V_B dominate the set E in G; that is, every e in E has a neighbor in V_R (to which it is connected in R) and a neighbor in V_B (to which it is connected in B). Thus, we can define a 2-coloring of H; the vertices in V_R are colored red, and those in V_B (as well as all other vertices if there are any) are colored blue. It is easy to see that this is a proper 2-coloring, completing the proof of the claim and hence that of the *NP*-completeness. ■

Other variants of the problem can be proved to be *NP*-complete in the same manner. For example:

1) Given a graph G with a root u, are there $c_G(u)$ edge-disjoint spanning trees in G of depth 2 rooted at u? It suffices to add one additional vertex of degree 2, and connect it only to the vertices r and b, and the proof above will still hold since $c_G(u) = 2$?

2) Given a graph G with a root u and an integer h, are there two edge-disjoint spanning trees in G of depth h rooted at u? It suffices to replace u by a path of length $h - 2$, $u = u_0, u_1, \dots, u_{h-2}$ where u_i is connected to u_{i+1} by two edges for $i = 0, \dots, h-3$, and where u_{h-2} is connected to the vertices of V and r and b as was u in the proof of Theorem 3. If one does not want double edges we have to distinguish two cases. If $h > 3$, then replace in the preceding construction each u_i, $i = 1, \dots, h-3$ by two vertices u_i and v_i, and connect u_i and v_i with u_{i+1} and v_{i+1}. If $h = 3$ then we do a complete different construction. We use again the graph G of the proof of Theorem 3 with its root u, and the four vertices r_1, r_2, b_1, b_2, of the remark in Theorem 2. Then we add, on the edges between u and each of these four vertices, four new vertices $r'_1\ r'_2\ b'_1\ b'_2$ and connect these four vertices by a complete graph.

3) Given a graph G with a root u, and an integer p, are there p edge-disjoint spanning trees in G of depth 2 rooted at u? This problem can be reduced to the hypergraph colorability problem with p colors, which is also *NP*-complete. Similarly, one can show the *NP*-completeness of finding $c_G(u)$ (or p) edge-disjoint spanning trees of depth h rooted at u.

The *NP*-completeness of the oriented problem can be obtained in a similar manner by replacing in the proof each edge by two symmetric arcs.

Acknowledgement

J-CB was supported by research program C3 and by DIMACS. PF was supported by research program C3 and by the Direction des Recherches, Etudes, et Techniques. This work was done while both authors were visiting the School of Computing Science, Simon Fraser University, Burnaby BC, CANADA. The authors are grateful to all the persons who helped during the preparation of this manuscript, in particular Noga Alon and Burkhard Monien. J-CB expresses his thanks to the organizers of Graph Theory Day 22; in particular, Gary Bloom for inviting him to the conference and thereby enabling him to run the New York Marathon 22.

References

[1] S.M. Hedetniemi, S.T. Hedetniemi and A.L. Liestman; A survey of gossiping and broadcasting in communication networks, *Networks*, **18**, 319--349 (1986).

[2] P. Fraigniaud and E. Lazard; Methods and problems of communication in usual networks, *Discrete Applied Math.*, special issue on broadcasting, submitted.

[3] J-C. Bermond, P. Hell, A. Liestman and J. Peters; Sparse broadcast graphs, *Discrete Applied Math.*, (1992) to appear.

[4] L.H. Khachatrian and O.S. Harutounian; Construction of new classes of minimal broadcast networks, *Proceedings of the Conference on Coding Theory*, Armenia (1990).

[5] M.J. Dinneen, M.R. Fellows and V. Faber; Algebraic constructions of efficient broadcast networks, in *Proceedings of Applied Algebra, Algebraic Algorithms and Error Correcting Codes*, **9**, *Lecture Notes in Computer Science*, **539**, 152-158 (1991).

[6] W. Knodel; New Gossips and telephones, *Discrete Mathematics*, **13**, 95 (1975).

[7] J-C. Bermond and P. Fraigniaud; Broadcasting and gossiping in deBruijn networks, *SIAM Journal on Computing* (1991) submitted.

[8] J. Edmonds; Edges-disjoint branchings, combinatorial algorithms, in *Combinatorial Algorithms*, {edited by R. Rustin}, Algorithmics Press, New York, 91-96 (1972).

[9] L. Lovász; On two minimax theorems in graph theory; *J. Combinatorial Theory*, **B21**, 96-103 (1976).

[10] S.L. Johnsson and Ching-Tien Ho; Optimum broadcasting and personalized communication in hypercubes, *IEEE Trans. Comp.*, **38** 1249-1268 (1989).

[11] J-C. Bermond, P. Michallon and D. Trystram; Broadcasting in wraparound meshes with parallel monodirectional links, *Parallel Computing*, to appear.

[12] P. Michallon, D. Trystram, and G. Villard; Optimal broadcast on wraparound meshes, *Journal of Parallel and Distributed Computing*, submitted.

[13] L. Lovász; Covering and coloring of hypergraphs; in *Proceedings of 4th Southeastern Conference on Combinatorics, Graph Theory, and Computing*, Utilitas Mathematica Publishing, Winnipeg, 3-12, (1973).